WHAT IS CLAIMED IS:

1. A software authentication system for authenticating a communication channel between a plurality of software elements, comprising:

a host computer having host storage including a first software element;

a second software element to authenticate said first software element;

wherein in response to said second software element making a request to said first software element for authentication of the first software element, the second software element retrieves a first encrypted digital signature from said host computer, the second software element retrieves a public key for use with said first encrypted digital signature and the second software element accesses at least one portion of a component stored in said host storage, said at least one portion of the accessed component is hashed to form a second digital signature;

wherein in response to receiving said encrypted first digital signature, said second software element decrypts said encrypted first digital signature with said public key and said second software element compares the first digital signature to the second digital signature; and

whereupon the occurrence of a correlation between said first and second digital signatures, the first software element is authenticated.

2. A software authentication system according to claim 1, wherein the first software element is a driver software element.

3. A software authentication system according to claim 2, wherein the second software element is a content providing software application, such that the content providing software application authenticates the driver software element.

4. A software authentication system according to claim 1, wherein the second software element instantiates a third software element which accesses said at least one portion of a component stored in said host storage in place of said accessing by said second software

component.

5. A software authentication system according to claim 4, wherein the third software element is instantiated in memory space allocated for drivers in said host computing system, and said first software element is a driver software element.

6. A software authentication system according to claim 1, wherein the component stored in host storage is the first software element.

7. A software authentication system according to claim 6, wherein the access of the first software element stored in host storage is during runtime of the software authentication system.

8. A software authentication system according to claim 1, wherein the component stored in host storage is a file stored on a hard drive of said host.

9. A software authentication system according to claim 1, further comprising a data storage device, wherein the communications channel between the first software element and the data storage device is authenticated with a technique including at least handshaking algorithms with a secure memory with authentication integrated circuit included in said data storage device.

10. A software authentication system according to claim 1, wherein the first software element is a driver software element.

11. A software authentication system according to claim 10, wherein the second software element is firmware included in a data storage device, such that the firmware included in the data storage device authenticates the driver software element.

12. A software authentication system according to claim 1, wherein the first software element is a content providing software application.

13. A software authentication system according to claim 12, wherein the second software element is a driver software element, such that the driver software element authenticates the content providing software application.

14. A software authentication system according to claim 1, wherein the second software element performs a handshaking algorithm with the first software element before proceeding to authenticate said first software element.

15. A software authentication system according to claim 1, further comprising a storage medium and a data storage device, wherein a communications channel between the data storage device and the storage medium is authenticated with a technique including at least one of a retroreflective marker, latent illuminance marker, disk indelible utility mark (DIUM), holographic marker included on said storage medium.

16. A software authentication system according to claim 1, wherein said hashed result is formed from said accessed component using at least one of few, division, multiplication, variable string addition, variable string exclusive-or and double variable string exclusive-or hash function algorithms.

17. A software authentication system according to claim 1, wherein said asymmetric encryption and decryption are performed using at least one of RSA, Diffie-Hellman, Elliptic-Curve and PGP asymmetric cryptography algorithms.

18. A software authentication system according to claim 1, wherein said correlation occurs

when one from the following group occurs: (1) when said first and second digital signatures are identical, (2) when a portion of said first digital signature is identical to a portion of second digital signature, (3) when said first digital signature equals said second digital signature after applying a predetermined algorithm to one of said first and second digital signatures and (4) when said first digital signature maps to said second according to an interpreted off-set match.

19. A method for authentication of a first software element stored in the memory of a host computer by a second software element, comprising:

said second software element requesting authentication information from said first software element;

in response to said requesting of authentication information, transmitting a first encrypted digital signature from said host computer to said second software element, retrieving by said second software element a public key for use with said first encrypted digital signature, accessing at least one portion of a component stored in the host storage, hashing said at least one portion to form a second digital signature;

in response to receiving said transmitted encrypted first digital signature, decrypting said encrypted first digital signature by said second software element with the public key and comparing the decrypted first digital signature to the second digital signature that is accessible to said second software element; and

determining that said first software element is authenticated if said first digital signature correlates with said second digital signature.

20. A method for authenticating according to claim 19, wherein the first software element is a data storage device driver software element, wherein the second software element is a content providing software application, and wherein the method for authentication is a method for authentication of the data storage device driver software element by the content providing software application.

21. A method for authenticating according to claim 19, further including instantiating a third software element which performs said accessing of said at least one portion of a component stored in said host storage.

22. A method for authenticating according to claim 21, wherein said instantiating includes instantiating said third software element in memory space allocated for drivers in said host computing system, and wherein said first software element is a driver software element.

23. A method for authenticating according to claim 19, wherein the accessing and hashing of said at least one portion of the component stored in host storage includes accessing and hashing at least one portion of the first software element.

24. A method for authenticating according to claim 23, wherein the accessing of at least one portion of the first software element stored in host storage is performed during runtime of the first software element.

25. A method for authenticating according to claim 19, wherein the accessing and hashing of said at least one portion of the component stored in host storage includes the accessing and hashing of a file stored on a hard drive of said host.

26. A method for authenticating according to claim 19, further comprising authenticating a communications channel between the first software element and a data storage device with a method including at least handshaking algorithms with a secure memory with authentication integrated circuit included in said data storage device.

27. A method for authenticating according to claim 19, wherein the first software element

stored in the host storage is a data storage device driver software element, wherein the second software element is a data storage device, and wherein the method for authentication is a method for authentication of the storage device driver software element by the data storage device.

28. A method for authenticating according to claim 19, further comprising authenticating the communications channel between a data storage device and a storage medium with a method including attaching at least one of a retroreflective marker, latent illuminance marker, disk indelible utility mark (DIUM), holographic marker to said storage medium.

29. A method for authenticating according to claim 19, wherein said hashing to form a hashed result from said accessed at least one portion of the component utilizes at least one of few, division, multiplication, variable string addition, variable string exclusive-or and double variable string exclusive-or hash function algorithms.

30. A method for authenticating according to claim 19, wherein said asymmetric encrypting and decrypting are performed using at least one of RSA, Diffie-Hellman, Elliptic-Curve and PGP asymmetric cryptography algorithms.

31. A method for authenticating according to claim 19, wherein said determining includes determining that said first software element is authenticated if one from the following group occurs: (1) if said first and second digital signatures are identical, (2) if a portion of said first digital signature is identical to a portion of second digital signature, (3) if said first digital signature equals said second digital signature after applying a predetermined algorithm to one of said first and second digital signatures and (4) if said first digital signature maps to said second according to an interpreted off-set match.

32. A computer-readable medium having computer-executable instructions for instructing a

computer to perform the method recited in claim 19.

33. A modulated data signal carrying computer-executable instructions for performing the method as recited in claim 19.

34. A method for authentication of a device driver software element stored in the memory of a host computer by an application, comprising:

said application instantiating a proxy driver software element;

said proxy driver software element requesting authentication information from said device driver software element;

in response to said requesting of authentication information, transmitting a first encrypted digital signature from said host computer to said application, retrieving by said application a public key for use with said first encrypted digital signature, accessing by said proxy driver software element at least one portion of a component stored in the host storage, hashing said at least one portion to form a second digital signature;

in response to receiving said transmitted encrypted first digital signature, decrypting said encrypted first digital signature by said application with the public key and comparing the decrypted first digital signature to the second digital signature that is accessible to the application via said proxy driver software element; and

determining that said device driver software element is authenticated if said first digital signature correlates with said second digital signature.

35. A method for authenticating according to claim 34, wherein the accessing and hashing of said at least one portion of the component stored in host storage includes accessing and hashing at least one portion of the device driver software element.

36. A method for authenticating according to claim 35, wherein the accessing of at least one

portion of the device driver software element stored in host storage is performed during runtime of the device driver software element.

37. A method for authenticating according to claim 34, wherein the accessing and hashing of said at least one portion of the component stored in host storage includes the accessing and hashing of a file stored on a hard drive of said host.

38. A method for authenticating according to claim 34, further comprising authenticating a communications channel between the device driver software element and a data storage device with a method including at least handshaking algorithms with a secure memory with authentication integrated circuit included in said data storage device.

39. A method for authenticating according to claim 34, further comprising authenticating the communications channel between a data storage device and a storage medium with a method including attaching at least one of a retroreflective marker, latent illuminance marker, disk indelible utility mark (DIUM), holographic marker to said storage medium.

40. A method for authenticating according to claim 34, wherein said hashing to form a hashed result from said accessed at least one portion of the component utilizes at least one of few, division, multiplication, variable string addition, variable string exclusive-or and double variable string exclusive-or hash function algorithms.

41. A method for authenticating according to claim 34, wherein said asymmetric encrypting and decrypting are performed using at least one of RSA, Diffie-Hellman, Elliptic-Curve and PGP asymmetric cryptography algorithms.

42. A method for authenticating according to claim 34, wherein said determining includes

determining that said device driver software element is authenticated if one from the following group occurs: (1) if said first and second digital signatures are identical, (2) if a portion of said first digital signature is identical to a portion of second digital signature, (3) if said first digital signature equals said second digital signature after applying a predetermined algorithm to one of said first and second digital signatures and (4) if said first digital signature maps to said second according to an interpreted off-set match.

43. A computer-readable medium having computer-executable instructions for instructing a computer to perform the method recited in claim 34.

44. A modulated data signal carrying computer-executable instructions for performing the method as recited in claim 34.